

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-implemented method for executing a program comprising a function call and one or more subsequent instructions, the method comprising [[the]] steps of:
processing, on a first thread, a function defined by the function call, the function having one or more at least two programmer predefined typical return values;
for each predefined return value, pre-processing on an additional thread, the one or more subsequent instructions which are associated with a given predefined return value, wherein said pre-processing is performed concurrent with the processing of the first thread; and assuming that the function returned that pre-defined return value;
thereby enabling said processor, on completion of processing said function, to make use of the pre-processing completed by the additional thread which used the actual return value responsive to completion of the processing of said function on said first thread, wherein said function returns an actual return value upon said completion, continuing processing of the additional thread associated with the actual return value while terminating at least one other additional thread that is not associated with the actual return value, such that the one or more subsequent instructions associated with the actual return value are speculatively executed prior to (i) the completion of the processing of said function and (ii) the return of the actual return value by said function.
2. (Original) The method of claim 1, wherein the program comprises a plurality of subsequent instructions defining one or more additional functions, the method further comprising:
pre-processing on each additional thread the plurality of subsequent instructions until a function is reached which is of external effect; and
blocking on said function having external effect until the actual return value is determined by the first thread.
3. (Original) The method of claim 2, wherein the blocking step also blocks on reaching a function which is affected by an external event.
4. (Currently Amended) An apparatus for executing a program comprising a function call and one or more subsequent instructions, the apparatus comprising:

means for processing, on a first thread, a function defined by the function call, the function having one or more at least two predefined typical return values;

means for pre-processing for each predefined return value, on an additional thread, the one or more subsequent instructions which are associated with a given predefined return value, wherein said pre-processing is performed concurrent with the processing of the first thread; and assuming that the function returned that pre-defined return value,

thereby enabling said processor, on completion of processing said function, to make use of the pre-processing completed by the additional thread which used the actual return value

means, responsive to completion of the processing of said function on said first thread, wherein said function returns an actual return value upon said completion, for continuing processing of the additional thread associated with the actual return value while terminating at least one other additional thread that is not associated with the actual return value, such that the one or more subsequent instructions associated with the actual return value are speculatively executed prior to (i) the completion of the processing of said function and (ii) the return of the actual return value by said function.

5. (Original) The apparatus of claim 4, wherein the program comprises a plurality of subsequent instructions defining one or more additional functions, the apparatus further comprising:

means for pre-processing on each additional thread the plurality of subsequent instructions until a function is reached which is of external effect; and

means for blocking on said function having external effect until the actual return value is determined by the first thread.

6. (Original) The apparatus of claim 5, wherein the blocking means is operable to also block on reaching a function which is affected by an external event.

7. (Currently Amended) A computer program comprising computer-readable program code means adapted to perform, when said program is run on a computer, the method of claim 1.

8. (Currently Amended) A compiler for generating the computer-readable program code means of Claim 7 a computer program comprising a function call defining a function, having one or more programmer predefined typical return values, and one or more subsequent instructions, the compiler comprising means for generating executable code, said executable code for instructing a computer to process on a first thread the function and to pre-process, for each defined typical return value, on an additional thread the one or more subsequent instructions assuming that the function returned that pre-

~~defined return value, thereby enabling said processor, on completion of processing said function, to make use of the pre-processing completed by the additional thread which used the actual return value.~~

9. (New) An apparatus for executing a program comprising a function call and a plurality of subsequent instructions, the apparatus comprising:

means for processing, on a first thread, a function defined by the function call, the function having a plurality of predefined typical return values;

means for pre-processing for each predefined return value, on an additional thread, certain ones of the subsequent instructions which are associated with a given predefined return value until a non-restricted function having an external effect is encountered, wherein said pre-processing is performed for each additional thread concurrent with the processing of the first thread; and

means, responsive to completion of the processing of said function on said first thread, wherein said function returns an actual return value upon said completion, for continuing processing of the additional thread associated with the actual return value while terminating (1) other additional threads that are not associated with the actual return value and (2) the first thread, such that the certain ones of the subsequent instructions associated with the actual return value are speculatively executed prior to (i) the completion of the processing of said function and (ii) the return of the actual return value by said function.